A Systematic Task and Knowledge-Based Process to Tune Cybersecurity Training to User Learning Groups: Application to Email Phishing Attacks

Nathan Monsoro, Célia Martinie, Philippe Palanque, Théo Saubanère

A Systematic Task and Knowledge-Based Process to Tune Cybersecurity Training to User Learning Groups: Application to Email Phishing Attacks

Nathan Monsoro[1[0009-0000-5263-3967]], Célia Martinie[1[0000-0001-7907-3170]], Philippe Palanque[1[0000-0002-5381-971X]] and Théo Saubanère[1 2[0009-0006-2856-7537]]

[1] University Toulouse III – Paul Sabatier, 31400 Toulouse, France
{nathan.monsoro, celia.martinie, philippe.palanque, theo.saubanere}@irit.fr

[2] Guiana Space Centre, CNES, 97310 Kourou, France
theo.saubanere@cnes.fr

Abstract. Cybersecurity training is one of the most important countermeasures to address cybersecurity threats and their reported increase in terms of types and occurrences. Several approaches addressing the development of cybersecurity training have been proposed but a careful analysis of these approaches highlighted limitations both in terms of identification of required knowledge, skills, in terms of description of users' tasks (the job they have to perform) as well as in terms of adaptation of the training to diverse user groups. This paper proposes a systematic process to tune cybersecurity training for diverse user groups, and in particular to support the development of cybersecurity training programs for different learning groups (built from the analysis of the diverse user groups). We illustrate this process on the concrete case of phishing attacks.

Keywords: cybersecurity, training, task models, user, context.

1 Introduction

Cybersecurity training is a widely studied topic (167,000+ entries in Google Scholar on March 21st 2024) as it is identified as one of the most important countermeasures to address cybersecurity threats by helping users identifying and preventing their undesired effects. Many scientific contributions (e.g. [22]) point out that "ordinary" users are considered the weakest link. This is referred to as the weakest link phenomenon [27], and training is the recommended means to deal with this phenomenon [27]. The use of cybersecurity training is widespread in various organizations [5], and our work focuses on systematic methods to develop training programs for employees of such organizations. Many approaches to the development of cybersecurity training have been proposed [21] (e.g. game-based, presentation-based, simulation-based…). The development of these cybersecurity training programs requires the identification of contextual knowledge [26] because part of the knowledge to acquire belongs to real-world

contexts and depends on the organization and the actual work of the users [12]. Contextual knowledge decomposes into several types of knowledge, for which we use the following explicit distinctions [20]: procedural knowledge, declarative knowledge, situational knowledge, and strategic knowledge. Procedural knowledge embeds information about how to perform a task. An example of procedural knowledge is the sequence of user tasks to authenticate on a mobile device (i.e. recall PIN code, enter PIN code, validate, etc.). Declarative knowledge embeds information users know and which can be true or false. An example of declarative knowledge for authentication on an internet bank account portal would be "the need for a PIN code and its value". Beyond these two types of knowledge there are two additional ones: strategic knowledge and situational knowledge [9]. Strategic knowledge gathers the multiple sets of procedural and declarative knowledge that enable a user to reach a goal (e.g. user may choose different strategies to authenticate e.g. fingerprint or PIN code (if both are offered by the system)). Situational knowledge gathers multiple sets of procedural and declarative knowledge that relates to a particular situation (e.g. if the authentication finger got hurt, the user should decide the PIN code authentication). Situational and strategic types of knowledge are tightly coupled to real-world contexts.

Another limitation of existing approaches for the development of cybersecurity training is that they do not take into account the variability of users. Addressing this, requires users to be marshaled into user groups [11].

Last limitation we propose to address is to describe precisely and exhaustively the tasks that each user must perform [21]. Indeed, threats will interfere with users' tasks and addressing the threats requires a complete understanding of the people's work and the time and place threats might appear within these tasks. Beyond, these user tasks need to be amended to explicitly represent how threats should be processed by the users.

We propose a systematic task and knowledge-based process to tune cybersecurity training for different user groups with different needs in terms of knowledge acquisition.

This paper is organized as follows. The second section presents the related work on the approaches to training, as well as on cybersecurity training. The third section presents the proposed systematic and knowledge-based process to identify learning groups and to tune cybersecurity training for these learning groups. The fourth section presents the results of the application of this process to the tuning of cybersecurity training on the specific case of phishing attacks. The fifth section elaborates on the main possibilities that this task and knowledge-based process enables. The last section concludes the paper and identifies paths for future work.

2 Related work

The main benefits of using systematic approaches to training are that they offer a stepwise ordered process to design and develop training programs, which helps to ensure that training goals are reached, as well as enables its exact replicability. Such systematic approaches to training have been applied to diverse domains such as (military, nuclear powerplant operator, air traffic management and even cybersecurity [5], just to name a

few). Even though various cybersecurity approaches to training have been proposed, most of them are complementary to and compatible with systematic approaches to training program development [5].

2.1 Instructional System Design and the ADDIE Framework

ADDIE is an Instructional System Development (ISD) founded in 1975 for the need for a Systematic Approach to Training (SAT) to encompass a military training problem back then [1]. Initially proposed in the Interservice Procedures for Instructional Systems Development (IPISD) [3, 4] this ISD evolved. It decomposes into five phases [1]: Analysis, Design, Development, Implementation, and Evaluation. We detail hereafter the main objectives of the Analysis and Design phases because they are the main phases the proposed process supports. These objectives are labeled with an identifier (e.g. A1, A2…) to which we will refer later in the paper.

The Analysis phase aims to:

- **A1**: Identify all the tasks the operator needs to know to be able to fulfill thoroughly his job,
- **A2**: Highlight, within the previously listed, the tasks done on the job by the operators,
- **A3**: Collect the job performance measure required to qualify the operator at the end of the training, for each task selected,
- **A4**: Analyze the courses already available, to determine if they are fully or partly usable,
- **A5**: Choose the most suitable instructional setting, for each task selected.

The Design phase aims to:

- **D1**: Convert tasks from the analysis phase into learning objectives that ensure the operator's mastery of the job will be obtained,
- **D2**: Prepare how to evaluate each learning objective,
- **D3**: Identify the entry behavior (skills and knowledge prerequisites for the trainees) and test on a sample of trainees to verify if assumptions were correct,
- **D4**: Structure and sequence the training tasks

The Development phase mainly aims to develop the materials (**Dev1**) and make the course ready for implementation (**Dev2**). The Implementation phase aims to execute the training plan with trainees. The Evaluation Phase aims to analyze the training effectiveness and to produce revisions of it, if needed.

The ADDIE phases are to be applied in sequence but the entire process is iterative, meaning that once the Evaluation phase is over, the outcome of this phase feeds back into the Analysis phase of the next iteration of the process, which ends when the training program and training material meets the training needs.

2.2 Cybersecurity training

The cybersecurity community tackles the topic of user training in diverse ways [21]: game-based, presentation-based, simulation-based, information-based, video-based, text-based, and discussion-based. However, several contributions (detailed below) point out that existing approaches do not take into account adequately the specificities in terms of knowledge and skills of multiple user groups and do not detail enough the user tasks. Jampen et al. [15] highlight that "*a lack of individualization of training limits the efficiency of training*". Stockett [24] recommends to "*structure your training to your audience! Some groups attending training are going to have a better grasp of technology and information security than others*". Kävrestad et al. [11] also confirm this view and argue that there is little methodological support to develop training according to user needs: "*…user should be trained in different ways. However, sources detailing which those groups are and how they should be trained are scarce*". Chowdury et al. [5] proposed a cybersecurity training framework based on a revised version of the ADDIE model, bringing the Evaluation phase to the center of the process and conducting evaluation throughout the life cycle of the training development. In addition, they argue for the personalization of training according to individual learner profiles. However, they do not propose a concrete technique to implement such a recommendation. Our contribution is a step towards the personalization of training. It is complementary to the ADDIE-based approaches.

3 A systematic task and knowledge-based process to tune cybersecurity training to user learning groups

The proposed process supports the Analysis and Design phases of an ADDIE-based cybersecurity training. It enables the analysis of users' tasks that have an impact on cybersecurity, as well as the identification of the required learning units and learning groups. It also makes explicit the knowledge that is needed to be acquired.

3.1 Overview of the process

Fig. 1 presents a diagrammatic view of the process. The topmost tasks are the preliminary tasks, that are to be performed before the process itself. The output of these tasks is used during the next phases of the process. The next phase of the process is the analysis phase (steps 1 to 3), which is performed for each threat identified by the expert's knowledge, and for each prerequisite identified. The last phase of this process is the design phase (steps 4 to 7). In this phase, the knowledge is split into learning units which are temporally ordered based on their prerequisites. The process produces learning groups and learning units based on the knowledge level of the trainees using task models. These task models represent the declarative and procedural knowledge that the trainees have to learn. These task models can later (in the development phase of ADDIE) be used to create the training material for each learning group.

3.2 Preliminary steps

At the top of **Fig. 1**, are presented the preliminary steps of the process.

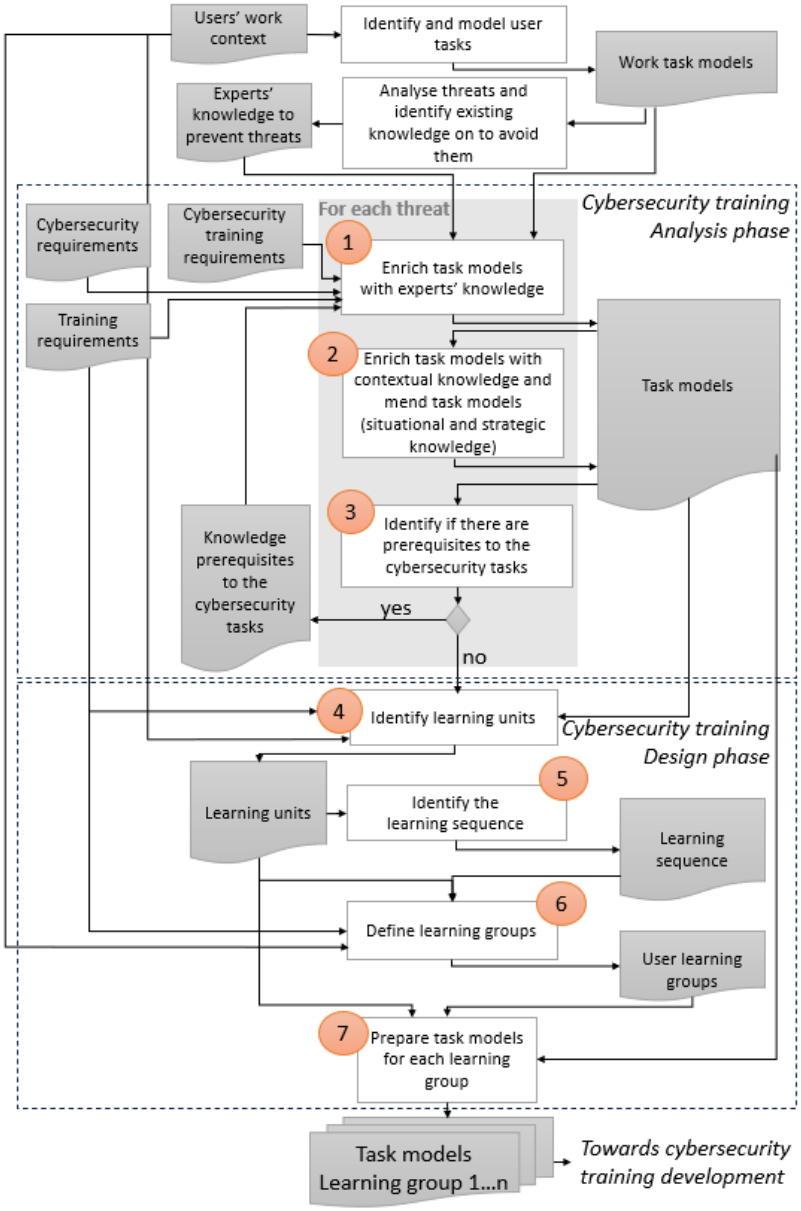

Fig. 1. A systematic task and knowledge-based process to tune cybersecurity training

These steps might not be performed by the organization planning the training, but the outcomes of these tasks are necessary to the proper conduction of the following steps

of the process. The users' work context is an input to these preliminary steps. They are mandatory information required to apply the proposed process. Task models are descriptions of user tasks. They are graphical representations of the work the users perform with an interactive application or system [18]. They consist of a hierarchical (structured in terms of goals, sub-goals, and actions) and temporally ordered description of user activities. They also enable the description of the required information and knowledge to perform user tasks. We use task models because they enable the systematic identification of procedural and declarative knowledge, as well as situational and strategic knowledge [20]. They also enable the description of how users interact with security mechanisms [16]. The presented process is also applicable using textual task descriptions in replacement of task models, though in that case, it may be harder to systematically identify the required knowledge for the cybersecurity training.

3.3 Main Steps of the Process

Steps 1, 2, and 3 of the process are performed for each threat identified during the preliminary steps of the process.

Step 1. Enrich Task Models with Experts' Knowledge. The first step aims to refine user work task models using experts' knowledge about identified threats. The outcome is a set of context-independent task models that contain the knowledge required to prevent or mitigate the identified threats. They gather several types of knowledge: procedural as well as declarative knowledge. This step supports applying the two first objectives of the ADDIE Analysis phase: A1 and A2. Several sets of job tasks may interleave with tasks related to the handling of cybersecurity threats.

Step 2. Identify contextual knowledge and Mend Task Models. The second step aims to identify context-dependent knowledge and mend the task models accordingly. In particular, it aims to identify the strategic and situational knowledge sub-types for both procedural and declarative knowledge (as defined in the introduction). The outcome of this step is a set of context-dependent task models including the description of strategic and situational knowledge. This step also supports applying the two first objectives of the ADDIE Analysis phase: A1 and A2.

Step 3. Identify prerequisite. The third step aims to browse systematically the context-dependent task models to identify procedural and declarative knowledge that the user needs to know to be able to prevent the identified threat. If procedural knowledge (which corresponds to a set of branches in the task model) requiring refinement is identified (labeled "Knowledge prerequisites to the cybersecurity tasks"), then steps 1-2-3 are performed again for each of these particular tasks, until there are no more prerequisites identified. The outcome of this step is the complete set of context-dependent task models including the description of strategic and situational knowledge. This step also supports applying the two first objectives of the ADDIE Analysis phase: A1 and A2.

Step 4. Identify learning units. The fourth step aims to extract parts of the complete set of context-dependent task models into learning units. Each task model is browsed to identify the sub-goal(s), their associated sub-trees, and declarative knowledge (strategic and situational) that belong to a learning unit. This step requires the training requirements and users' work context because the selection of procedure and declarative knowledge that belong to a learning unit depends on the training requirements (e.g. "training sessions should not last more than 1 hour and a half") and on users' context (e.g. abilities, initial skillset, availabilities…). The outcome of this step is a list of learning units associated with a set of procedural and declarative knowledge. This step supports the application of the following objectives of the ADDIE Design phase: D1 and D4.

Step 5. Identify the learning sequence. The fifth step aims to identify and describe temporal ordering constraints between learning units. These temporal ordering constraints are based on the learning progression between the learning units. The outcome of this step is a learning sequence for all of the learning units. This step supports the application of the following objective of the ADDIE Design phase: D4.

Step 6. Define learning groups. The sixth step aims to cluster sets of learning units according to training requirements (e.g. "every employee has to follow the training") and users' work context (e.g. users' prior knowledge, users' training record, users' missions in the organization…). Each cluster of learning units belongs to a learning group. The outcome of this step is a list of learning groups with their associated learning units temporally ordered in a learning sequence. This step also supports the application of the following objective of the ADDIE Design phase: D4.

Step 7. Prepare task models for each learning group. The seventh step aims to associate relevant parts of task models with learning groups. In particular, for one learning group, each learning unit is associated with the task model or part of the task model that describes the tasks (procedural, declarative, situational, and strategic knowledge) to be learned during the learning unit. The outcomes of this step are the sets of context-dependent task models, each set being associated with a learning group. This step supports the application of the objectives Dev1 and Dev2 of the ADDIE Development phase because the produced sets of task models enable the preparation of which tasks will have to be explained to the different learning groups. In particular, they support the production of training scenarios [14] [17] [19].

3.4 Level of expertise and learning groups

Each learning group corresponds to one level of expertise. The learning group for which all of the context-dependent task models have been selected corresponds to the group of novices. The individuals belonging to this group have every task and associated procedural and declarative knowledge to learn. The learning for which the shortest set of

context-dependent task models has been selected corresponds to the group with the highest level of expertise. Usually, learning groups are marshaled in three groups: novice, intermediate and experts. The benefit of applying a task and knowledge description for learning group identification is that those usually abstract groups are made concrete (in terms of tasks they perform and knowledge they need) supporting the evaluation of the quality of the training program (but this is outside the scope of the current paper).

4 Illustrative example: the tuning of cybersecurity training to phishing attacks for several user groups

To demonstrate the applicability of the process to concrete cybersecurity training, we present in this section an illustrative example addressing email phishing threat applied to two user groups: users who have no skills and knowledge about the technical settings of email clients and internet browsers, and users who have skills and knowledge about it. Such user groups can be found in organizations, such as companies or universities, and training entry tests may be used to identify in which group a person belongs to. We selected the email phishing threat as 2023 is identified as the worst recorded year of email phishing with almost five million of attacks observed [2]. Thus, training the users to identify and counter email phishing is of prime importance. Email phishing is a social engineering technique that involves forging the identity of a trustworthy source (be it in an email [25], by imitating a website [7], or in many more ways such as those described in [8]), and tricking the target into revealing personal information.

In this section, we present the results of the application of the two main phases of the proposed process: "Cybersecurity training **analysis phase**" and "Cybersecurity training **design phase**". In particular, we present the task model of the main goals and task model of the sub-goals "detection of a phishing email" (**Fig. 2**), an extract of the context-dependent task model for the sub-goal of "analyzing the sender's intent" (**Fig. 3**). We also present the tables produced after the identification of learning units and learning groups (merged in **Table 1**), and the diagram of the learning sequence for both identified user groups (**Fig. 4**).

For the purpose of this process, the task models are realized thank to the HAMSTERS notation. The interested readers can find more about the HAMSTERS notation in [18].

4.1 Step 1. Enrich task models with Experts' knowledge

To model the tasks of our users, we use the three-stage sequence of tasks to detect phishing emails as identified by IT experts [2, 23]. These three stages are face value, suspicion, and decision. **Fig. 2** presents the task model of the high-level goals and sub-goals to detect a phishing email. The goal "Detect a phishing email" decomposes in a sequence of first consulting the email (abstract task labeled "Consult the email"), then inspecting the email looking for information (abstract task labeled "Inspect the email…"), and then dealing with the email (abstract task labeled "Deal with the email"). The subtask "Inspect the email looking for information" refines in interleaving the tasks

of inspecting the sender, checking for the potential use of manipulation techniques, inspecting the attachments, inspecting the redirections, inspecting the form and style, and analyzing the sender's intent and the email's information (last row of tasks). The first sub-goal of the task model (abstract task "Consult the email") is composed of work tasks and no security tasks, however, its description is included in the task model because this sub-goal produces the information that will be required for the security tasks of the later stages of the process. The second sub-goal ("Inspect the email looking for information") is composed exclusively of security tasks. The third sub-goal ("Deal with the email") is the work of the user with the addition of a few security tasks.

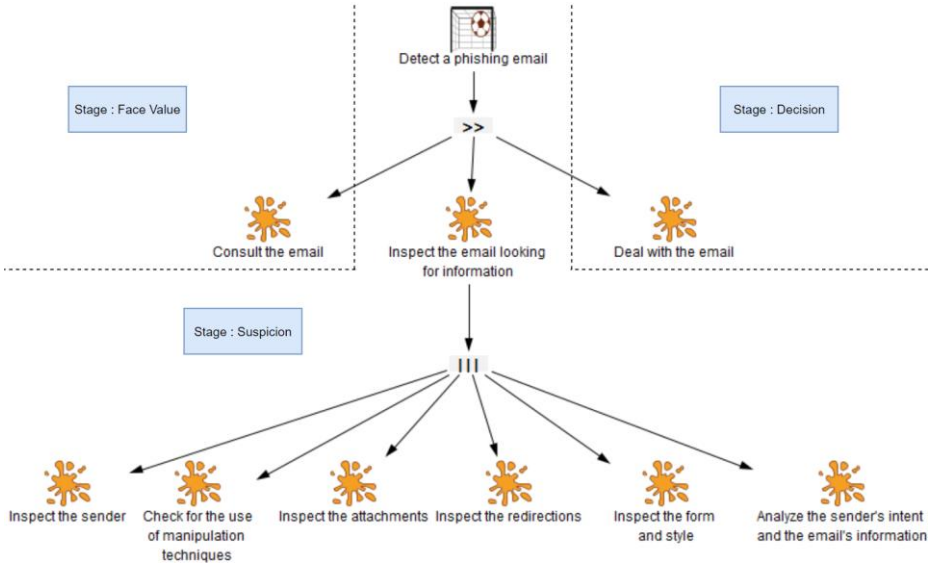

Fig. 2. The main goal and sub-goals of the task model of detection of a phishing email

4.2 Step 2. Identify contextual knowledge and mend task models

Fig. 3 presents the task model of the subgoal "Analyze the sender's intent and the email's information" (at the top of the tree). The subgoal is refined in a concurrency between the subtasks "Analyze if the metadata of the email are unusual" and "Does the intent of the email match the user's expectations". The former decomposes in a concurrency between the sequence of checking the date of the email (perceptive task labeled "Check the date of the email") and then analyzing the date of the expedition of the email (cognitive task labeled "Analyze the date the expedition of the email"), and analyzing the list of receivers (cognitive task labeled "Analyze the list of receivers"). The second subtask decomposes in a concurrency between the tasks of analyzing the sense made by the action required by the sender, thinking about the best communication option the sender had, thinking about the implication of the email in the sender's life, and checking if the sender opened a ticket. Checking if the sender opened a ticket is strategic

procedural knowledge because we consider that in the context of our organization, creating a ticket is common. **Fig. 3** also includes the knowledge and information objects the information and declarative knowledge objects are linked to the task they take part in. The "Inf" objects correspond to information the user acquires while performing the tasks, the other objects represent the declarative knowledge. The "DK" objects represent the declarative knowledge that is not refined, the "Stk" objects represent the declarative strategic knowledge and the "SiK" objects represent the declarative situational knowledge. An arrow going from an object to a task means that the task consumes this object (as an example, to analyze if the action required makes sense, the information contained in the email is necessary and is used to perform the task) and an arrow going from a task to an object means that the task produces this information/knowledge. The blue arrow with the "St" annotation in the task model going from the "concurrent" operator to the task "check if the sender opened a ticket" (the rightmost task of **Fig. 3**) provides information on procedural knowledge. Here "St" stands for strategic knowledge because this task is context-specific.

4.3 Step 3. Identify prerequisites

We identified the prerequisites from the context-dependent task models. An example of a prerequisite is "Inspect the attachments" (the 3rd task on the bottom line of **Fig. 2**). Users may or not know how to inspect an attachment. This task has thus to be described in detail, to support the training of users who are not familiar with this task.

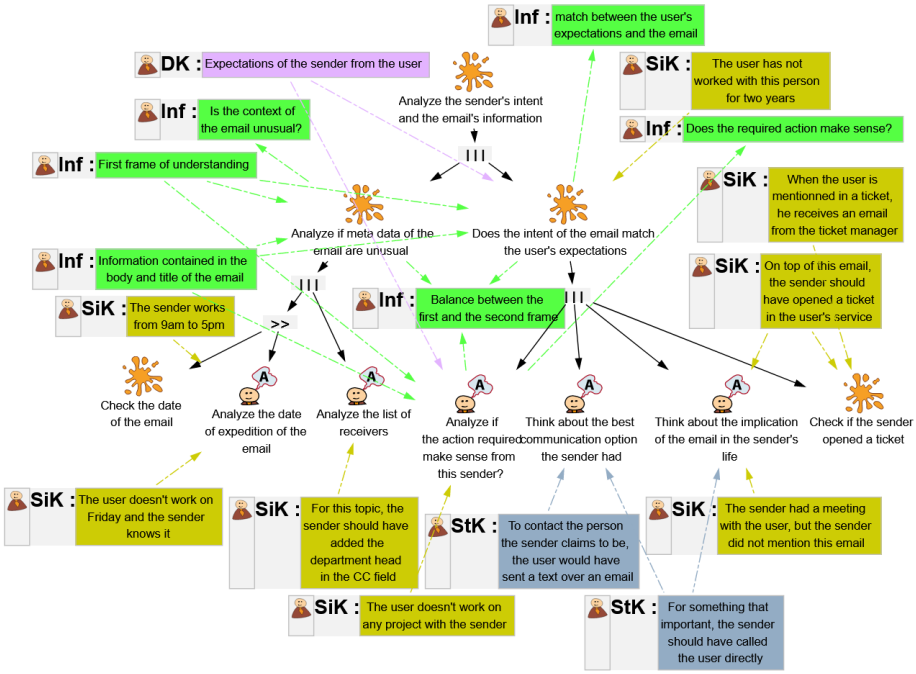

Fig. 3. Extract of the context-dependent task model for the sub-goal of analyzing the sender's intent and the email's information

The three first columns of **Table 1** present the knowledge identified from the task model required to complete the training.

4.4 Step 4. Identify learning units

Identifying the learning units is at the discretion of the people responsible for designing the training process. It might vary depending on the learning needs of the users, the span of the training, or the availability of the trainers.

Table 1. Merging of the tables produced during step 3 (white background) and step 4 (grey)

Prerequisite	Associated type of knowledge	Context dependent	Learning Units ID	Prerequisite LU for learning (ID)
Consult the email	Procedural	No	LU 1	4 5
Inspect the email looking for information	Procedural	Yes	LU 2	1 6 7 8 9 11 12 13 14 16 17 18 19 20
Deal with the email	Procedural	Yes	LU 3	2 9 10
Look for attachment	Procedural	No	LU 4	None
What is an email address	Declarative	No	LU 5	None
What is a hyperlink	Declarative	No	LU 6	None
Access the details of the sender's information	Procedural	No	LU 7	None
Check the destination of a hyperlink	Procedural	No	LU 8	6
Check the extension of an attachment	Procedural	No	LU 9	None
Deal with a phishing email	Procedural	No	LU 10	15
Known aspects of human psychology that can be taken advantage of	Declarative	No	LU 11	None
What are file extensions and what are their risks	Declarative	No	LU 12	None
What is a domain	Declarative	No	LU 13	None
What is a secure domain	Declarative	No	LU 14	13
What is a phishing email	Declarative	No	LU 15	None
The hierarchy and responsibilities of the other employees	Declarative	Yes	LU 16	None
The way the ticket manager works	Declarative	Yes	LU 17	None
How to check if the user received a ticket	Procedural	Yes	LU 18	17
Known emotions that can interfere with the user's rational thinking	Declarative	Yes	LU 19	None
How to identify manipulation techniques	Procedural	No	LU 20	19 11

For the sake of exhaustivity and precision in the explanations of the results of the implementation of the process, we decided that each prerequisite and each objective would be a learning unit. The fourth column of **Table 1** shows the ID of the different learning units and the title of the learning units is the description of the prerequisite. The fifth column in **Table 1** records the prerequisite learning units of a learning unit, i.e. before learning what a secure domain is, the users need to learn what a domain is altogether. The identification of the prerequisites was done using the task models.

4.5 Step 5. Identify the learning sequence

The learning sequence may vary according to the schedule of the trainees and the span of the learning units, as well as the settings of training chosen for the learning units.

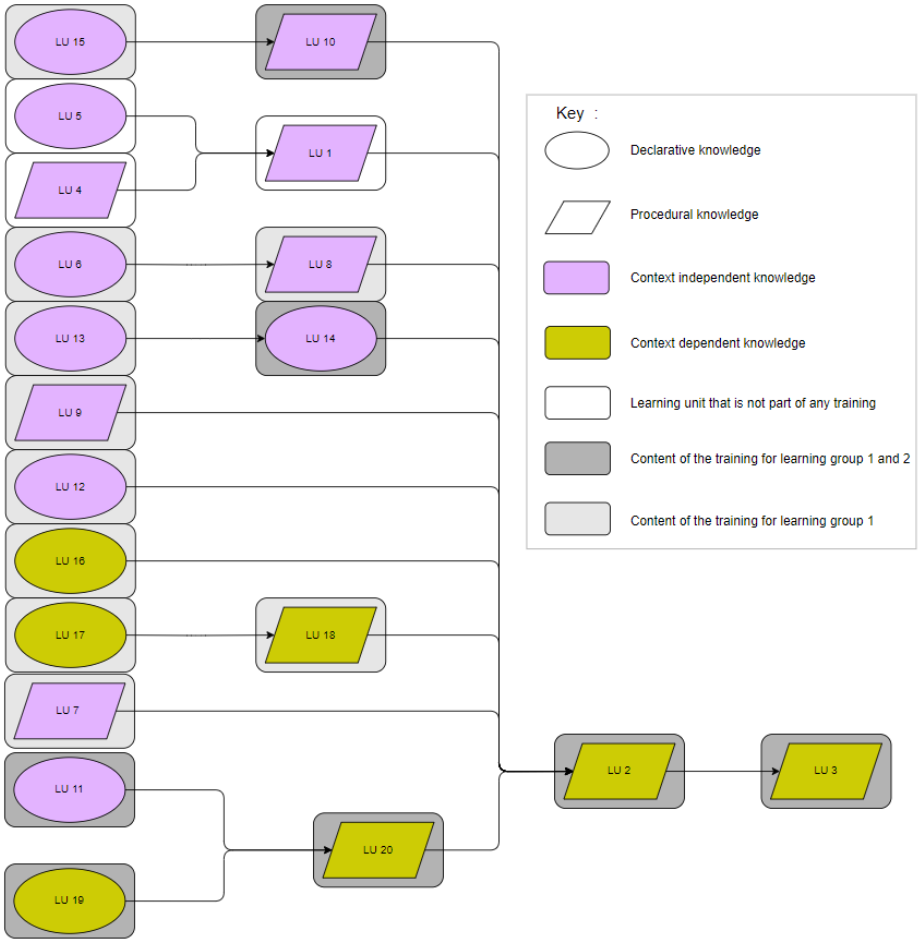

Fig. 4. Extract of the learning sequence for the email phishing detection training

In this example, we produced the learning sequence (in **Fig. 4**) using only the temporal constraints between the identified learning units. **Fig. 4** also highlights the type of knowledge to be acquired for each unit, which supports the development of adequate training means and materials.

4.6 Step 6. Define learning groups

Our two main user groups are users who have no skills and knowledge about the technical settings of email clients and internet browsers (group 1), and users who have skills and knowledge about it (group 2). All of the learning units are required for users of Group 1. We identified the learning units that are not required for the second user group. In **Fig. 4**, the learning groups are presented associated with the corresponding required learning units for each group. For example, the learning unit LU 13 "What is a domain" is associated with learning group 1, but not with learning group 2. Some of the LU are not part of any learning group because they are work-related and are considered acquired for every trainee.

4.7 Step 7. Prepare task model for each learning group

The last step of our process creates a set of task models for each learning group including all the knowledge they have to acquire. The task model of each learning group consists, in our case of the task model "detect a phishing email" (goal of **Fig. 3**) since this is the topic of the training. The difference between each group's task model will reside in its refinement. Since the trainees in the second group of our case study already know how to check if the sender has opened a ticket (the rightmost task in **Fig. 3**), the task model of the second group will not detail how to check if the sender opened a ticket. However, since the trainees of the first learning group need to acquire this knowledge, this task will be refined in their task model. The declarative knowledge can be added when necessary (when not known by the user group) to the task model. We have shown the results from the application of the process The interested reader can find all the information about the task models and produced artifacts on the webpage *https://sites.google.com/view/haisa-2024-submission-11-websi/home-page*.

5 Conclusion and future work

We presented a process for the systematic and knowledge-based tuning of cybersecurity training for user groups with different needs in terms of knowledge acquisition. This process integrates with systematic approaches to training and supports the analysis, design, and development of cybersecurity training programs. We presented the results of the application of the process on cybersecurity training for phishing attacks and highlighted how the learning units can be shaped according to user groups and training requirements. The identification of learning units and learning groups enables us to determine the top level of expertise, as well as to prepare the path to bring everyone up to the top competence level. This transition from novice to expert also requires to take

into account human performance. Cockburn et al. [6] identified four domains to improve human performance to support the transition from novice to expert for tasks that are supported by interactive systems: *intramodal*, *intermodal*, *vocabulary extension*, and *task mapping*. The two first domains relate to the performance in using an interactive system and the two last relate to the user knowledge, and strategic knowledge in particular for the *task mapping* domain. Taking into account the in-use interactive system for the training development is very important and the proposed process thus integrates with training development approaches for interactive systems [17].

Beyond taking into account the user groups and training requirements, the shaping, grouping, and ordering of learning units may require including the level of criticality of user tasks and system functions, the role of the users in the organization, as well as the cultural context of the trainees [10]. For example, the preparation of dedicated and duplicated learning units could help ensure that users reach a particular retention level for the knowledge related to these tasks and functions. The user groups of the same population are bound to change according to the trainer's and the trainees' availability, and so are the learning units.

This work presented a systematic user-centered process to analyze training needs and design training sequences, groups, and tasks, where the key inputs are the work task models of the trainees. Despite the proposed process being systematic, we emphasize the need for customization of the user groups and learning unit according to the needs of the organization.

Acknowledgments. This work was supported partially by the France 2030 ANR Project ANR-23-PECL-0009 TRUSTINCloudS.

References

1. Allen, W., C. Overview and Evolution of the ADDIE Training System. Advances in Developing Human Resources, 8(4), 430-441, 2006. https://doi.org/10.1177/1523422306292942
2. APWG, Phishing activity trends report, 4th quarter 2023, last accessed March 2024, https://docs.apwg.org/reports/apwg_trends_report_q4_2023.pdf
3. Branson, R. K. The Interservice Procedures for Instructional Systems Development. Educational Technology, 18(3), 11–14, 1978. http://www.jstor.org/stable/44418942
4. Branson, R. K. Interservice procedures for instructional systems development: Executive summary and model. Tallahassee, Fla.: Center for Educational Technology, Florida State University, 1975, https://apps.dtic.mil/sti/citations/tr/ADA019486
5. Chowdhury, N., Katsikas, S., Gkioulos, V. Modeling effective cybersecurity training frameworks: A delphi method-based study, Computers & Security, Volume 113, 2022, 102551, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2021.102551.
6. Cockburn, A., Gutwin, C., Scarr, J., Malacria, S. 2014. Supporting Novice to Expert Transitions in User Interfaces. ACM Comput. Surv. 47, 2, Article 31 (January 2015), 36 pages.
7. Dhamija, R., Tygar, J. D., Hearst, M. 2006. Why phishing works. Proc. of CHI '06. ACM, New York, NY, USA, 581–590. https://doi.org/10.1145/1124772.1124861
8. Gupta, S., Singhal, A., Kapoor, A. A literature survey on social engineering attacks: Phishing attack. Proc. of ICCCA 2016, 537-540, https://doi.org/10.1109/CCAA.2016.7813778

9. de Jong, T., & Ferguson-Hessler, M. G. (1996). Types and qualities of knowledge. Educational psychologist, 31(2), 105-113
10. Karimnia, R., Maennel, K., & Shahin, M. (2022, February). Culturally-sensitive Cybersecurity Awareness Program Design for Iranian High-school Students. In ICISSP (pp. 121-132).
11. Kävrestad, J., Fallatah, W., Furnell, S. (2023). Cybersecurity Training Acceptance: A Literature Review. HAISA 2023, vol 674. Springer, Cham.
12. Kävrestad, J., Furnell, S., & Nohlberg, M. (2024). User perception of Context-Based Micro-Training – a method for cybersecurity training. Information Security Journal: A Global Perspective, 33(2), 121–137. https://doi.org/10.1080/19393555.2023.2222713
13. International Standard Organization. ISO 9241-11:2018. Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts. 2nd edition, 2018.
14. Lallai, G., Zedda, G. L., Martinie, C., Palanque, P., Pisano, M., Spano, D. 2021. Engineering Task-based Augmented Reality Guidance: Application to the Training of Aircraft Flight Procedures. Interact. Comput. 33, 1 (2021), 17–39.
15. Jampen, D., Gür, G., Sutter, T. et al. Don't click: towards an effective anti-phishing training. A comparative literature review. Hum. Cent. Comput. Inf. Sci. 10, 33 (2020). https://doi.org/10.1186/s13673-020-00237-7
16. Martinie, C., Naqvi, B. (2023). On using the Task Models for Validation and Evolution of Usable Security Design Patterns. In HAISA 2023, vol 674. Springer, Cham. https://doi.org/10.1007/978-3-031-38530-8_32
17. Martinie, C., Navarre, D., Palanque, P., Barboni, E., Steere, S. 2022. Engineering Operations-based Training. Proc. ACM Hum.-Comput. Interact. 6, EICS, Article 164 (June 2022), 25 pages. https://doi.org/10.1145/3534518
18. Martinie, C., Palanque, P., Barboni, E. 2022. Principles of Task Analysis and Modeling: Understanding Activity, Modeling Tasks, and Analyzing Models. Handbook of Human Computer Interaction. Springer, Cham.
19. Martinie, C., Palanque, P., Navarre, D., Winckler, M., Poupart, E. 2011. Model-Based Training: An Approach Supporting Operability of Critical Interactive Systems. Proc. EICS 2011, ACM, 53–62. https://doi.org/10.1145/1996461.1996495
20. Martinie, C., Palanque, P., Ragosta, M., Fahssi, R. 2013. Extending procedural task models by systematic explicit integration of objects, knowledge and information. Proc. of ECCE 2013. ACM, Article 23, 1–10. https://doi.org/10.1145/2501907.2501954
21. Prümmer, J., van Steen, T., van den Berg, B. 2024. A systematic review of current cybersecurity training methods. Computers & Security, Volume 136, 103585, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2023.103585.
22. Sasse, M.A., Brostoff, S. & Weirich, D. Transforming the 'Weakest Link' — a Human/Computer Interaction Approach to Usable and Effective Security. BT Technology Journal 19, 122–131 (2001). https://doi.org/10.1023/A:1011902718709
23. Staggs, J., Beyer, R., Mol, M., Fisher, M. Brummel, B. J., Hale, J. 2014. A perceptual taxonomy of contextual cues for cyber trust. Proc. of CISSE, 152–169, https://cisse.info/journal/index.php/cisse/article/view/9/CISSE_v02_i01_a01.pdf
24. Stockett, J. 2018. Dr. InfoSec: How to Teach Your Community to Stop Worrying and Love 2-Factor Authentication. 2018. Proc. of SIGUCCS 2018, ACM, 21–23.
25. Wash, R. 2020. How Experts Detect Phishing Scam Emails. 2020. Proc. ACM Hum.-Comput. Interact. 4, CSCW2, Article 160, https://doi.org/10.1145/3415231
26. Workman, M. D., Luévanos, J. A., Mai, B. 2022. A Study of Cybersecurity Education Using a Present-Test-Practice-Assess Model. IEEE Transactions on Education, vol. 65, no. 1, pp. 40-45, https://doi.org/10.1109/TE.2021.3086025.

27. Yan, Z., Robertson, T., Yan, R., Park, S.Y., Bordoff, S., Chen, Q., Sprissler, E. 2018. Finding the weakest links in the weakest link: How well do undergraduate students make cybersecurity judgment? Com Hum. Beh., 84, 375-382.